Beyond Nash Equilibrium: Solution Concepts for the 21st Century

Joe Halpern

and many collaborators . . .

Cornell University

Nash equilibrium and security

- An often useful way to think of security is as a game between an adversary and the “good” participants in the protocol.
  - Allows us to model incentives of participants
  - Tradeoffs between costs of security and amount of security
- Game theorists understand games in terms of *solution concepts*
  - meant to describe what the outcome of a game will be
- *Nash equilibrium* (NE) is the most common solution concept.
  - A NE is a *strategy profile* (one strategy for each player) such that no player can do better by unilaterally deviating
  - Intuition: it’s a steady state of play (technically: a fixed point)
    - Each players holds correct beliefs about what the other players are doing and plays a best response to those beliefs.

The good news

The good news:

- Often, NE gives insight, and does predict what people do
- **Theorem:** [Nash] Every finite game has a Nash equilibrium (if we allow mixed (randomized) strategies).

The bad news

- NE gives quite unreasonable answers in a number of games
  - e.g., repeated prisoners' dilemma, discussed later
- How do agents learn what other agents are doing if the game is played only once!
  - What if there are multiple Nash equilibria?
    - Which one is played?
- Why should an agent assume that other agents will play their part of a NE, even if there is only one?
- What if agents are not aware of some aspects of the game
  - There may be lack of awareness of their moves, of other players' moves, or of who is playing the game

Alternative Solution Concepts

To deal with these problems, many refinements of and alternatives to NE have been considered in the game theory literature:

- rationalizability
- sequential equilibrium
- (trembling hand) perfect equilibrium
- proper equilibrium
- iterated deletion of weakly (or strongly) dominated strategies
- . . .

None of these address the concerns that I want to focus on.

New problems

- NE is not robust
  - It does not handle “faulty” or “unexpected” behavior
  - It does not deal with coalitions
- NE does not take computation costs into account
- NE assumes that the structure of the game is common knowledge
  - What if a player is not aware of some moves he can make?

k-Resilient Equilibria

NE tolerates deviations by one player.

- It's consistent with NE that 2 players could do better by deviating.

An equilibrium is k-*resilient* if no group of size k can gain by deviating (in a coordinated way).

Example: $n > 1$ players must play either 0 or 1.

- if everyone plays 0, everyone gets 1
- if exactly two players play 1, they get 2; the rest get 0.
- otherwise; everyone gets 0.

Everyone playing 0 is a NE, but not 2-resilient.

- Nash equilibrium = 1-resilient equilibrium.
- In general, k-resilient equilibria do not exist if $k > 1$.
- Aumann [1959] already considers resilient equilibria.
- But resilience does not give us all the robustness we need in large systems.

Following work on robustness is joint with Ittai Abraham, Danny Dolev, and Rica Gonen.

"Irrational" Players

Some agents don't seem to respond to incentives, perhaps because

- their utilities are not what we thought they were
- they are irrational
- they have faulty computers

Apparently "irrational" behavior is not uncommon:

- People share on Gnutella and Kazaa, seed on BitTorrent

Example: Consider a group of n bargaining agents.

- If they all stay and bargain, then all get 2.
- Anyone who goes home gets 1.
- Anyone who stays gets 0 if not everyone stays.

Everyone staying is a k-resilient Nash equilibrium for all $k < n$, but not immune to one “irrational” player going home.

- People certainly take such possibilities into account!

Immunity

A protocol is t-*immune* if the payoffs of “good” agents are not affected by the actions of up to t other agents.

- Somewhat like *Byzantine agreement* in distributed computing.
- Good agents reach agreement despite up to t faulty agents.

A (k, t)-*robust* protocol tolerates coalitions of size k and is t-immune.

- Nash equilibrium = (1,0)-robustness
- In general, (k, t)-robust equilibria don’t exist
  - they can be obtained with the help of *mediators*

Mediators

Consider an auction where people do not want to bid publicly

- public bidding reveals useful information
- don't want to do this in bidding for, e.g., oil drilling rights

If there were a mediator (trusted third party), we'd be all set . . .

- Distributed computing example: Byzantine agreement

Implementing Mediators

Can we eliminate the mediator? If so, when?

- Work in economics: implementing mediators with "cheap talk" [Myerson, Forges, . . .]
  - "implementation" means that if a NE can be achieved with a mediator, the same NE can be achieved without
- Work in CS: *multi-party computation* [Ben-Or, Goldwasser, Goldreich, Micali, Wigderson, . . .]
  - "implementation" means that "good" players follow the recommended protocol; "bad" players can do anything they like

By considering (k, t)-robust equilibria, we can generalize the work in both CS and economics.

Typical results

- If $n > 3k + 3t$, a (k, t)-robust strategy $\vec{\sigma}$ with a mediator can be implemented using cheap talk.
  - No knowledge of other agents' utilities required
  - The protocol has bounded running time that does not depend on the utilities.
  - Can't do this if $n \leq 3k + 3t$.
- If $n > 2k + 3t$, agents' utilities are known, and there is a *punishment strategy* (a way of punishing someone caught deviating), then we can implement a mediator
  - Can't do this if $n \leq 2k + 3t$ or no punishment strategy
  - Unbounded running time required (constant expected time).

- If $n > 2k + 2t$ and a broadcast facility is available, can ϵ-implement a mediator.
  - Can't do it if $n \leq 2k + 2t$.
- If $n \leq k + t$, assuming cryptography, polynomially-bounded players, a $(k + t)$-punishment strategy, and a PKI, then can ϵ-implement mediators using cheap talk.

Note how standard distributed computing assumptions make a big difference to implementation!

Bottom line: We need solution concepts that take coalitions and fault-tolerance seriously.

Making Computation Costly

Work on computational NE joint with Rafael Pass.

Example: You are given a number n-bit number x.

- You can guess whether it's prime, or play safe and say nothing.
  - If you guess right, you get $10; if you guess wrong, you lose $10; if you play safe, you get $1.
  - Only one NE in this 1-player game: giving the right answer.
    - Computation is costless
    - That doesn't seem descriptively accurate!

The idea of making computation cost part of equilibrium notion goes back to Rubinstein [1985].

- He used finite automata, charged for size of automaton used

A More General Framework

We consider *Bayesian games*:

- Each agent has a type, chosen according to some distribution
  - The type represents agent's private information (e.g., salary)
- Agents choose a Turing machine (TM)
- Associated with each TM M and type t is its *complexity*
  - The complexity of running M on t
- Each agent i gets a utility depending on the
  - profile of types, outputs ($M(t)$), complexities
    - I might just want to get my output faster than you

Can then define Nash Equilibrium as usual.

The good news

The addition of complexities allows us to capture important features:

- In the primality testing game, for a large input, you'll play safe because of the cost of computation
- Can capture overhead in switching strategies
- Can explain some experimentally-observed results.

Repeated Prisoner's Dilemma:

Suppose we play Prisoner's Dilemma a fixed number k times.

	C	D
C	$(3,3)$	$(-5,5)$
D	$(5,-5)$	$(-1,-1)$

- The only NE is to always defect
- People typically cooperate (and do better than "rational" agents who play NE)!

Suppose there is a small cost to memory and a discount factor $> .5$.

- Then *tit-for-tat* gives a NE if k is large enough
  - Tit-for-tat: start by cooperating, then at step $m+1$ do what the other player did at step m.
  - In equilibrium, both players cooperate throughout the game
- This remains true even if only one player has a cost for memory!

The bad news?

NE might not exist.

- Consider *roshambo* (rock-paper-scissors)
- Unique NE: randomize $1/3$–$1/3$–$1/3$
- But suppose we charge for randomization
  - deterministic strategies are free
- Then there's no NE!
  - The best response to a randomized strategy is a deterministic strategy

But perhaps this is not so bad:

- Taking computation into account should cause us to rethink things!

Redefining Protocol Security

Key Result: Using computational NE, can give a game-theoretic definition of security that takes computation and incentives into account

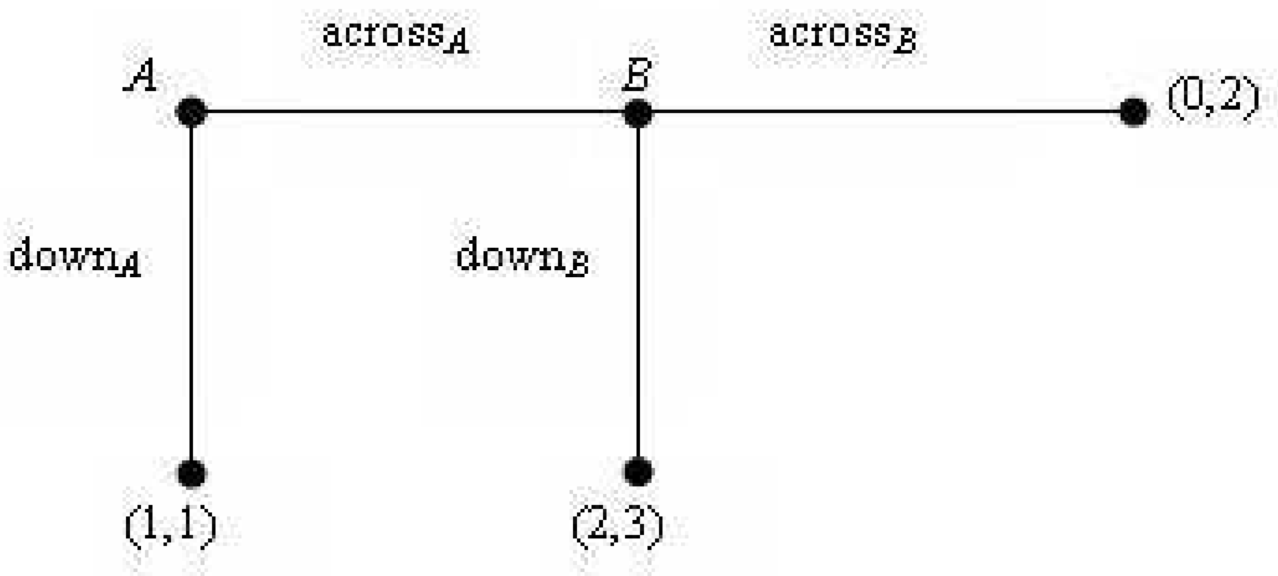
- Rough idea of definition: Π is a secure implementation of f if, for all utility functions, if it is a NE to play with the mediator to compute f, then it is a NE to use Π (a cheap-talk protocol)
- The definition does not mention privacy;
  - this is taken care of by choosing utilities appropriately
- Can prove that (under minimal assumptions) this definition is equivalent to *precise zero knowledge* [Micali/Pass, 2006]
  - Two approaches for dealing with "deviating" players are intimately connected: NE and zero-knowledge simulation

(Lack of) Awareness

Work on awareness is joint with Leandro Rêgo.

- Standard game theory models assume that the structure of the game is common knowledge among the players.
  - This includes the possible moves and the set of players
- **Problem:** Not always a reasonable assumption; for example:
  - war settings
    - one side may not be aware of weapons the other side has
  - financial markets
    - an investor may not be aware of new innovations
  - auctions in large networks,
    - you may not be aware of who the bidders are

A Game With Lack of Awareness



- One Nash equilibrium of this game
  - A plays across$_A$, B plays down$_B$ (not unique).
- But if A is not aware that B can play down$_B$, A will play down$_A$.

Representing lack of awareness

NE does not always make sense if players are not aware of all moves

- We need a solution concept that takes awareness into account!
- First step: represent games where players may be unaware
- Key idea: use *augmented games*:
  - An *augmented game based on an underlying standard game* Γ is essentially Γ and, for each history h an *awareness level*:
    - the set of runs in the underlying game that the player who moves at h is aware of
  - Intuition: an augmented game describes the game from the point of view of an omniscient modeler or one of the players.

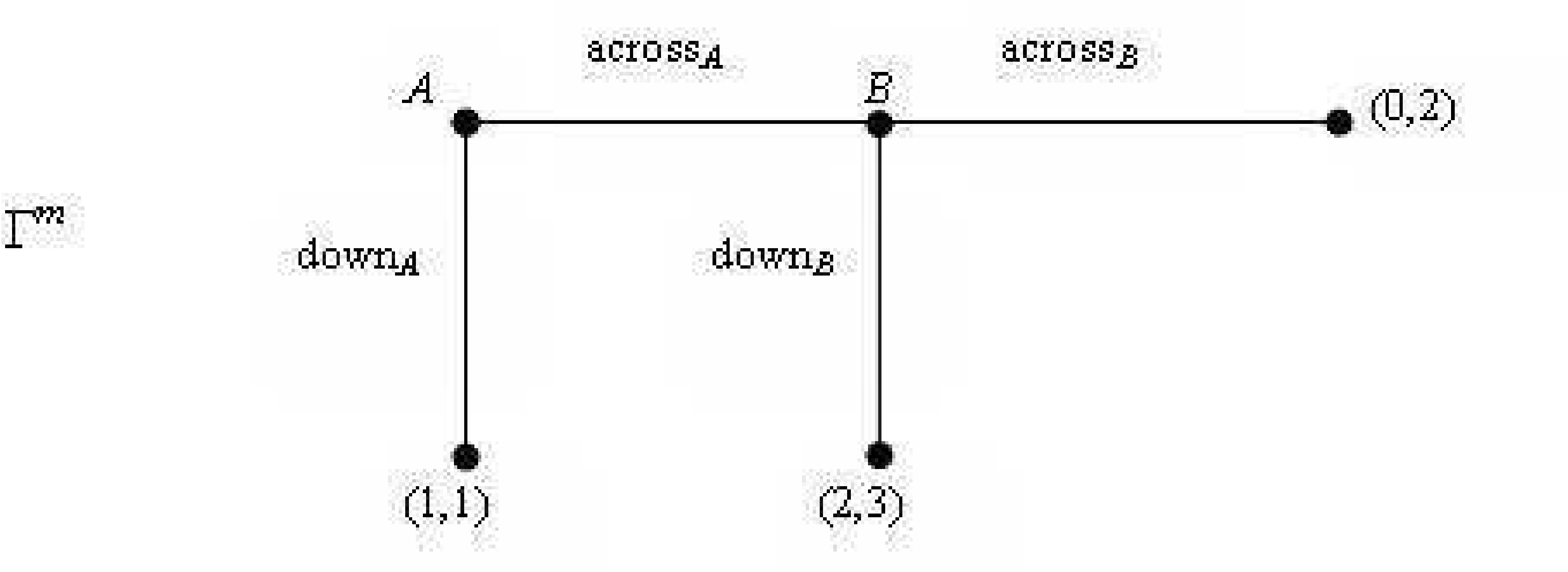
Augmented Games

Consider the earlier game. Suppose that

- players A and B are aware of all histories of the game;
- player A is uncertain as to whether player B is aware of run $\langle \text{across}_A, \text{down}_B \rangle$ and believes that B is unaware of it with probability p; and
- the type of player B that is aware of the run $\langle \text{across}_A, \text{down}_B \rangle$ is aware that player A is aware of all histories, and he knows A is uncertain about B's awareness level and knows the probability p.

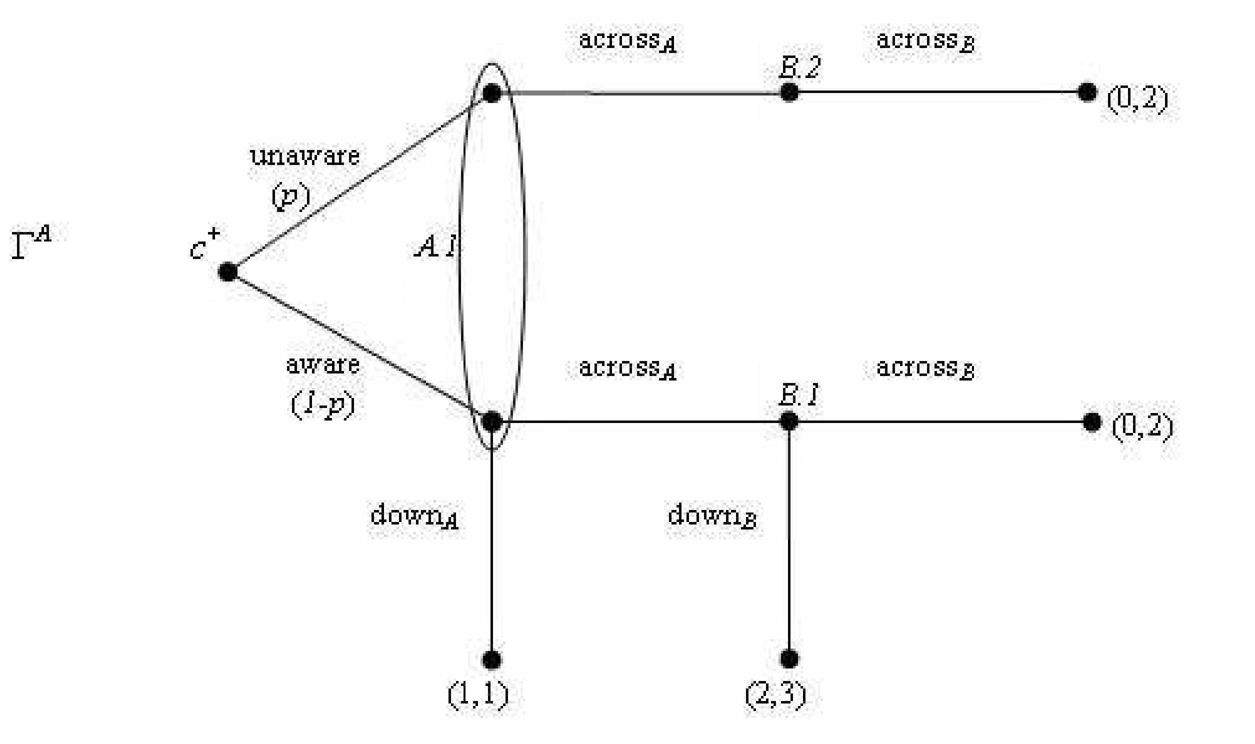
To represent this, we need three augmented games.

Modeler's Game



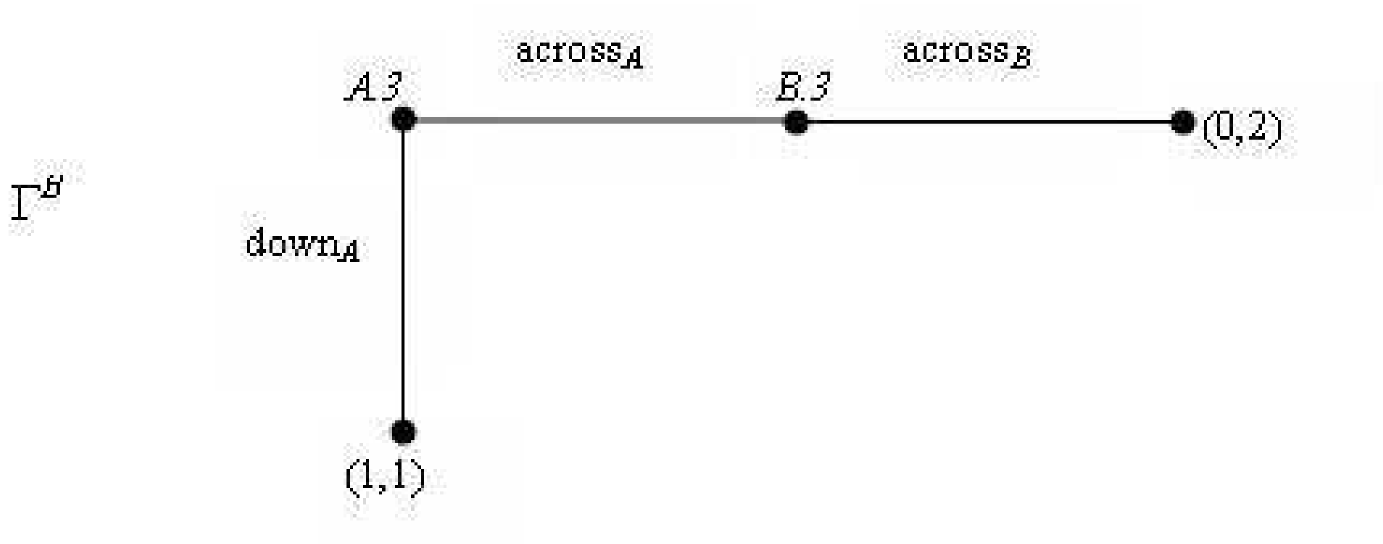
- Both A and B are aware of all histories of the underlying game.
- But A considers it possible that B is unaware.
  - To represent A's viewpoint, we need another augmented game.

A's View of the Game



- At node $B.2$, B is not aware of the run $\langle$across$_A$, down$_B\rangle$.
  - We need yet another augmented game to represent this.

(A's view of) B's view



- At node $A.3$, A is not aware of $\langle$across$_A$, down$_B\rangle$;
  - neither is B at $B.3$.
- **Moral:** to fully represent a *game with awareness* we need a set of augmented games.
  - Like a set of possible worlds in Kripke structures

Game with Awareness

A *game with awareness based on* Γ is a tuple $\Gamma^* = (\mathcal{G}, \Gamma^m, \mathcal{F})$, where

- $\mathcal{G}$ is a countable set of augmented games based on Γ;
- $\Gamma^m \in \mathcal{G}$ is an omniscient modeler's view of the game
- $\mathcal{F} : (\Gamma^+, h) \mapsto (\Gamma^h, I)$
  - h is a history in $\Gamma^+ \in \mathcal{G}$;
  - If player i moves at h in Γ^+ and $\mathcal{F}(\Gamma^+, h) = (\Gamma^h, I)$, then
    - Γ^h is the game that i believes to be the true game at h
    - I (i's *information set*) describes where i might be in Γ^h
      - I is the set of histories in Γ^h i considers possible;
      - histories in I are indistinguishable from i's point of view.

Local Strategies

- In a standard game, a strategy describes what a player does at each information set
- This doesn't make sense in games with awareness!
  - A player can't plan in advance what he will do when he becomes aware of new moves
- In a game $\Gamma^* = (\mathcal{G}, \Gamma^m, \mathcal{F})$ with awareness, we consider a collection of *local strategies*, one for each augmented game in $\mathcal{G}$
  - Intuitively, local strategy $\sigma_{i,\Gamma'}$ is the strategy that i would use if i thought that the true game was Γ'.
- There may be no relationship between the strategies $\sigma_{i,\Gamma'}$ for different games Γ'.

Generalized Nash Equilibrium

- Intuition: $\vec{\sigma}$ is a generalized Nash equilibrium if for every player i, if i believes he is playing game Γ', then his local strategy $\sigma_{i,\Gamma'}$ is a best response to the local strategies of other players in Γ'.
  - The local strategies of the other players are part of $\vec{\sigma}$.

Theorem: Every game with awareness has at least one generalized Nash equilibrium.

Awareness of Unawareness

Sometimes players may be aware that they are unaware of relevant moves:

- War settings: you know that an enemy may have new technologies of which you are not aware
- Delaying a decision: you may become aware of new issues tomorrow
- Chess: “lack of awareness” $\leftrightarrow$ “inability to compute”

Modeling Awareness of Unawareness

- If i is aware that j can make a move at h that i is not aware of, then j can make a "virtual move" at h in i's subjective representation of the game
  - The payoffs after a virtual move reflect i's beliefs about the outcome after the move.
    - Just like associating a value to a board position in chess
- Again, there is guaranteed to be a generalized Nash equilibrium.
- Ongoing work: connecting this abstract definition of unawareness to the computational definition

Related Work

- The first paper on unawareness by Feinberg (2004, 2005):
  - defines solution concepts indirectly, syntactically
  - no semantic framework
- Sequence of papers by Heifetz, Meier, Schipper (2005–08)
  - Awareness is characterized by a 3-valued logic
- Work with Rêgo dates back to 2005; appeared in AAMAS 2006
- Related papers on logics of awareness and unawareness
  - Fagin and Halpern (1985/88), Modica and Rusticchini (1994; 1999), . . . , Halpern and Rêgo (2005, 2006)
- *Lots* of recent papers, mainly in Econ:
  - 7 papers in TARK 2007, 6 papers in GAMES 2008

Conclusions

- I have suggested solution concepts for dealing with
  - fault tolerance
  - computation
  - (lack of) awareness
- Still need to take into account (among other things):
  - “obedient” players who follow the recommended protocol
    - Alvisi et al. call these “altruistic” players
  - “known” deviations: hoarders and altruist in a scrip system
  - asynchrony
  - computational equilibria in extensive form games
    - computation happens during the game